# Lab Report3

## Register File

### Design

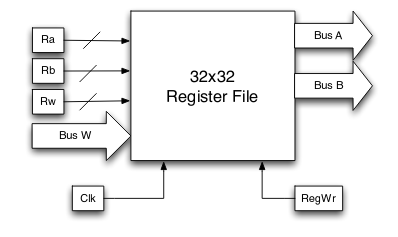


Fig 1.1 Register File Schematic

This register file has 32\*32 registers, 31 of them is writable, register 0 is always 0 to simulate r0 in actual processor. Output Bus A and Bus B are respectively allocated as output of Ra and Rb.

### Result

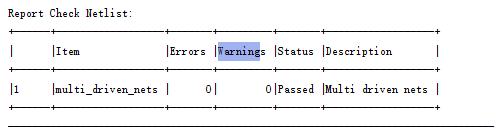


Fig 1.2 Synthesis Report

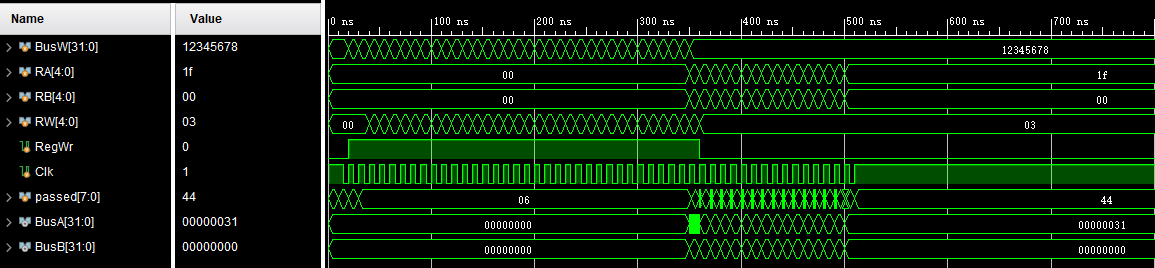


Fig 1.3 Register File Waveform



Fig 1.4 Register File output

## Data Memory

### Design

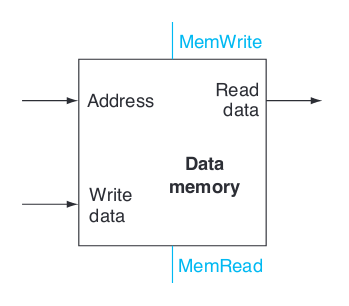


Fig 2.1 Data Memory Schematic

There is no input pin for write enable or read enable because writing and reading are impulsed by negedge and posedge.

### Result

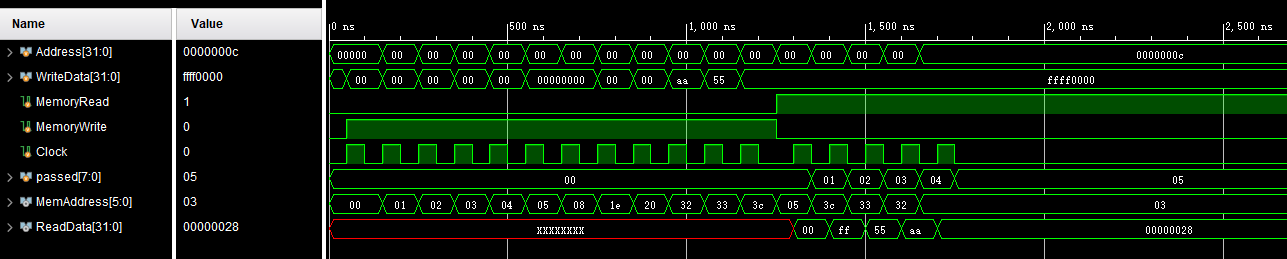


Fig 2.2 Wave form

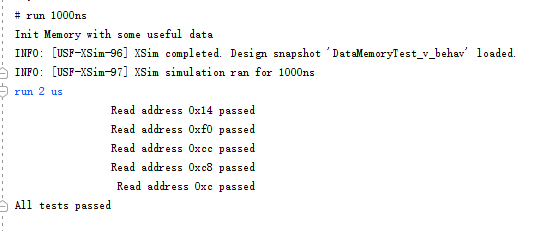


Fig 2.3 Logs

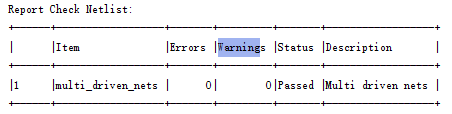


Fig 2.4 Report

## Questions

1. Yes. It could be done. By using dual-ported module, the size will be doubled as well. Considering the price of RAM and the speed of RAM and CPU, RAM is waiting for CPU orders, instead of the opposite way. Because memory is waiting for CPU commands for most of the time. Adding a not frequently used module is a waste. It’s not a good design.
2. MemRead is used when CPU needs to read data from memory. Yes it’s necessary. Because if there is no MemRead port, then the data will be continuously read from memory which cannot be accessed by CPU at once. On the other hand, the size of register file is much smaller than memory. Plus each unit has their own address, thus they can be accessed at once thus there is no need to set a switch(like MemRead) to control.
3. Synthesizable means the Verilog program can be corresponded into actual hardware connections. For example, ‘always @ xxx‘ is not synthesizable. Because it’s impossible to implement this phrase in hardware.